LastPass •••|

Technical Whitepaper

Summary of Key LastPass Security Principles

With solutions for single sign-on, password management, and multifactor authentication, LastPass helps consumers and businesses increase productivity and decrease the likelihood of password-related breaches.

Our security measures include:

- Reliable access to vault data: We endeavor to ensure you can manage and view your stored information whenever and wherever you need to, offline and online.
- Local-only encryption: LastPass is a host-proof solution, meaning the system is designed to ensure that only the user can access their data. Sensitive data is encrypted locally in a 'vault' that is stored on the end user's device and on our servers.
- **Centralized control for admins:** For businesses, LastPass provides a cloud-based admin dashboard for deployment and management. The dashboard includes configurable security policies, provisioning, reporting, and more.
- Security and encryption best practices: Sensitive data stored with LastPass is encrypted using a key that we never have. LastPass offers industry standard cryptography that is strong enough to defend against brute-force attacks.
- SOC-2 report: The Service Organization Control 2 (SOC 2) Type II attestation report is widely recognized as an information security "gold standard." Completing and maintaining the SOC 2 is just one way we demonstrate our commitment to data security, safeguarding of information and service availability.
- Top-level data centers: LastPass uses data centers in world-class hosting facilities that follow best practices for redundancy and stability.
- Timely incident response: Our team reacts quickly to investigate, verify, and resolve or mitigate reports of bugs or vulnerabilities. Our bug bounty program also incentivizes responsible disclosure and improvements to our service. Our product and customers benefit from the positive relationship we maintain with top security researchers.
- Regular audits and penetration tests: We engage trusted, world-class, third-party security firms to conduct routine audits and annual testing of the LastPass service and infrastructure.
- Backed by leading SaaS company LogMeIn: LogMeIn, Inc.'s category-defining products unlock the potential of the modern workforce by making it possible for millions of people and businesses around the globe to do their best work simply and securely—on any device, from any location, and at any time. A pioneer in remote work technology and a driving force behind today's work-from-anywhere movement, LogMeIn has become one of the

world's largest SaaS companies with tens of millions of active users, more than 3,500 global employees, over $1.3 billion in annual revenue, and approximately 2 million customers worldwide who use its software as an essential part of their daily lives. The company is headquartered in Boston, Massachusetts with additional locations in North America, South America, Europe, Asia, and Australia.

Table of Contents

Introduction

LastPass is helping people achieve effortless security, at home and in the workplace. As our business and personal worlds intersect on an increasing scale in our cloud-centric world, a strong foundation of secure authentication and access is critical to keeping systems, data, and assets safe.

Trusted by millions of consumers and tens of thousands of companies worldwide, LastPass safely stores passwords and grants access to the technology and services people rely on every day using single sign-on and password management.

Our core mission at LastPass is to keep customer information secure and provide a reliable service. This document shows how we accomplish this mission.

We help LastPass customers achieve better security in two ways:

1. Building security into the very foundation of the product, with additional layers of protection to safeguard customer data at all steps, and

2. Offering features, settings, and options that allow users and admins to customize LastPass to meet their specific security needs and follow best practices.

By building security and safeguards into the product, we strive to ensure that all LastPass users are protected from threats, both in the cloud and locally on their device.

And by offering configurable security features, we can equip end users and admins alike to minimize the poor security practices that put their private information at risk. With the implementation of LastPass' solutions, businesses and consumers can strengthen their defense against attackers.

We are constantly improving the LastPass software and updating our service with the latest technology as new attack vectors and security threats emerge. We work closely with members of the LastPass community and security researchers who help improve the service for the benefit of all users. LastPass fundamentally believes in taking proactive measures to review security reports, address issues, and regularly evaluate new technologies that will strengthen our security model.

Our Privacy Statement is available here (https://www.logmeininc.com/legal/privacy) and our Terms of Service are available here (https: www.logmeininc.com/legal/terms-and-conditions).

Logging in to the Password Manager

The Master Password

When a user creates an account for the LastPass password manager, they also create a Master Password. The Master Password is used to authenticate in to the LastPass account through the browser extension or by logging in to www.lastpass.com.

Once logged in, the user will be able to access and input the credentials for other websites that have been stored in the LastPass password manager. The vault is the space where a user can add, view, and manage credentials and other items that have been saved to LastPass. The vault is accessed by successfully entering the correct username and Master Password.

A strong Master Password

To ensure the security of their vault, it is essential that users choose a strong Master Password for their LastPass account. While we enforce industry-standard minimums when creating the Master Password, the user should make the Master Password as strong as possible. Specifically, that means a Master Password should be long and unique, with a mix of character types – it directly impacts the overall security of the data as other encryption keys are generated from this password. A moderately strong Master Password is also designed to ensure that a brute-force attack is unrealistic.

In our business solutions, admins can also enforce policies around the strength, complexity, and regular updates of Master Passwords, as well as prevent Master Password reuse.

The Master Password should never be used as a password for any other website or app. Even a variation of it should never be used for any other account. For example, a breach on another website could put a LastPass account at risk if a user re-uses their Master Password.

Users should also never share their Master Password with anyone, including LogMeIn employees. No one at LogMeIn, including our customer care team, ever needs to know the user's Master Password. Any requests to share a Master Password should be treated as a threat and reported to the LastPass team at https://support.logmeininc.com/lastpass/help/how-do-i-contact-customer-support-for-lastpass-lp010121 immediately.

Protecting the Master Password

The encrypted vault data in the LastPass password manager is meaningless to us and to anyone else without the decryption key, which is created from a combination of their username and Master Password. The Master Password is never sent to LastPass.

While the option to remember the Master Password is offered in the LastPass extension and mobile apps due to user demand, enabling it may reduce the security of the Master Password, and also makes it more likely that a user will forget it. Admins can enable a policy that prevents users from selecting the option to remember the Master Password.

New Device Verification

When a user logs in to their LastPass account from a new location and an unrecognized device, LastPass requires the user to complete a verification step to "trust" that new location/device.

The verification process involves LastPass sending a verification link to the email address used for a given LastPass account (or a designated security email address, if one has been added to the account). Once the user clicks the verification link, the new location/device is trusted.

The next time the user logs in from that device/location they will not be asked to complete the verification step.

Encryption Technology for the LastPass Password Manager

The encrypted vault for the LastPass password manager is designed to prevent the ability to decrypt a vault without a user's Master Password.

Local-Only Encryption Model

The LastPass password manager employs local-only encryption, also known as “host-proof hosting”. This type of solution is designed to allow only a LastPass user to decrypt and access their data. We call this “Local-Only Encryption”, which means that all sensitive vault data is encrypted and decrypted exclusively on the user’s local machine (such as Chrome, Firefox, iPhone, Android, the web vault, etc.), rather than after the data syncs to LastPass’ servers.

Only once data is encrypted with the user’s unique encryption key is the data synced to LastPass for secure storage. Sensitive data is transmitted to LastPass as a base64 encoded blob of encrypted data, and it never touches LastPass servers in a way that can be visible to LastPass. LastPass does not have access to nor does it store the Master Password, which prevents LastPass from having the ability to decrypt a user’s sensitive vault data.

This means that LastPass, and the employees who work here, can never access the sensitive data that a user stores in their vault nor can LastPass remotely access a user’s device. The data stored in LastPass is decrypted the instant it is needed on the user’s device, after the Master Password is successfully entered, including when the user is accessing their account via the web vault and any of the mobile apps.

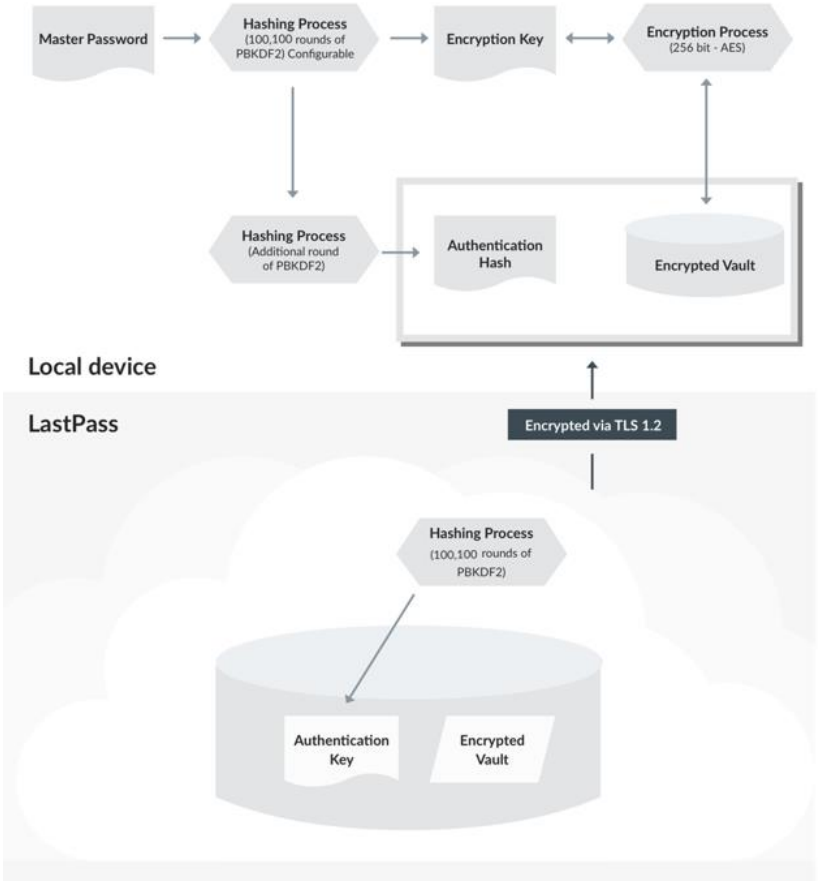
Account Lockout

LastPass also protects against brute-force attacks by locking accounts after repeated failed attempts to login. We regularly monitor accounts for signs of irregular or suspicious activity and will suspend accounts automatically when appropriate.

AES 256-bit Encryption

LastPass uses encryption and hashing algorithms of the highest standard to protect user data. Local-only encryption and locally-created, one-way salted hashes provide LastPass users with the best of both worlds: Complete security, with online accessibility, and syncing through the cloud.

LastPass encrypts user data with the trusted algorithm Advanced Encryption Standard (AES) in Cipher Block Chaining (CBC) mode with a 256-bit key generated from each user’s Master Password. The AES 256-bit algorithm is widely accepted as impenetrable and is the same military-grade encryption used by banks and the US military to secure Top Secret data. We believe that our best line of defense is simply not having access to unencrypted sensitive vault data.



Key Derivation

When a user creates their account, we first do a hash of the LastPass Master Password using the username as the salt. This is performed on the user's device (client-side).

We use a default of 100,100 rounds of PBKDF2-SHA256 to create the encryption key, on which we perform another single round of hashing, to generate the Master Password authentication hash (or the "login hash"). This hash is sent to the LastPass server so that we can perform an authentication check as the user is logging in. With that value, we use a salt (a random string per user) and do another 100,100 rounds of PBKDF2 hashing, in addition to hashing with Scrypt, a best-in-class hashing algorithm. When the user logs in, we compare this value to the authentication hash in our database. This is the value that LastPass stores on its servers to check against when the user next logs in.

The Master Password and encryption key are never sent to our servers. And because hashing is a one-way algorithm, LastPass cannot reverse the authentication hash that it receives.

In summary: With a good Master Password, cracking our algorithms is unrealistic, even for the strongest of computers.

Key Strengthening with PBKDF2

LastPass has implemented AES 256 with thousands of rounds of PBKDF2 SHA-256, a password-strengthening algorithm, to create the user's unique encryption key.

PBKDF2 is an adaptive one-way function which hashes a password multiple times with a hashing algorithm that can be chosen by the service provider. This makes it difficult for a computer to check that any one password is the correct Master Password during a brute-force attack.

LastPass has opted to use SHA-256, a slower hashing algorithm that provides more protection against brute-force attacks. LastPass performs x number of rounds of the function (100,100 by default) to create the encryption key, before a single additional round of PBKDF2 is done to create the user's login hash.

LastPass can increase the number of rounds over time to render brute-forcing the Master Password infeasible even as computers advance. Users also have the ability to increase the rounds of PBKDF2 in their account settings.

Increasing the number of iterations increases the work required to derive the hash. This makes verifying a password take longer, but in turn it also significantly increases the work needed to brute-force a password with a given hash.

The entire process is conducted client-side. The resulting login hash is what is communicated with LastPass. LastPass uses the hash to verify that the user is entering the correct Master Password when logging in to their account.

LastPass also performs 100,100 rounds of PBKDF2 server-side. This implementation of PBKDF2 client-side and server-side ensures that the two pieces of the user's data - the part that's stored offline locally and the part that's stored online on LastPass servers - are thoroughly protected.

PBKDF2 can be described as:

Derived Key = PBKDF2(PRF, Password, Salt, Iterations, Key Length)

Where:

PRF is the hash function to be used.

Password is the Master Password.

Salt consists of bits of data unique to each account used to ensure the same Master Password does not produce the same derived key.

Iterations is the desired number of iterations to run the PRF.

Key Length is the desired length of the derived key.

A vault encryption key is calculated with:

PBKDF2(SHA-256, Master Password, Username, 100,100, 256)

To create a login hash, an extra level of PBKDF2-SHA256 is run on the user's password using the user's vault encryption key to create another 256-bit hash, thus increasing the number of iterations to 100,101.

Login hash = PBKDF2(SHA-256, Master Password, Username, 100,101, 256)

This hash value is sent to LastPass and used for account authentication. Additional measures are taken to protect this hash before it is stored by LastPass, as described in the Login Hash Storage section of this white paper.

Due to the number of rounds used, users may notice slowness or problems connecting to LastPass when logging in via certain browsers, such as legacy versions of Internet Explorer.

LastPass Single Sign-On Solution Security

Although the Single Sign-On (SSO) product shares many core components with the overall LastPass infrastructure, we want to provide information about how LastPass delivers secure SSO and highlight elements that are unique to the SSO product.

SAML Single Sign-On

Security Assertion Markup Language (SAML) is an XML-based web security standard that is used to communicate user authentication and authorization information between an Identity Provider (IdP) and a Service Provider (SP). Once the authentication and authorization are successful, the user gains access to the services offered by the SP.

LastPass provides a SAML-based SSO integration with any web application that supports SAML. LastPass serves as an IdP, controls and manages all user identity information, and will authenticate and authorize the user on the SP side.

Enforcing Security Policies

LastPass solutions offers additional layers of control and protection to companies via the Admin Console. Admins can control the provisioning and de-provisioning of users, mandate the use of security features, and set organization-wide security policies that are customized for the unique needs of their corporate environment. A user kill switch ensures that departing or rogue employees can have their access revoked in real-time.

LastPass allows admins to audit employee password habits and see if employees are reusing passwords, reusing their Master Password, and putting the organization at risk through their actions.

Companies also benefit from detailed reporting logs for auditing and compliance purposes. In addition to the data encryption and storage benefits, LastPass solutions allow companies to create password policies and data breach prevention practices that are manageable and enforceable.

Learn more about policies here (https://support.logmeininc.com/lastpass/help/add-and-manage-enterprise-policies-lp080002).

Multifactor Authentication for LastPass Accounts

LastPass currently supports over a dozen multifactor authentication vendors, including LastPass MFA. Learn more about LastPass MFA at https://www.lastpass.com/products/multifactor-authentication and refer to the LastPass MFA section in this whitepaper.

LastPass encourages users to enable multifactor authentication to add an additional layer of protection to a LastPass account. Multifactor authentication requires another piece of information before access is granted.

Companies can mandate use of multifactor authentication with LastPass through policies available in the admin dashboard.

Multifactor authentication requires two or more authentication factors, including something the user knows (the Master Password), in addition to something they have (a code, a key) and/or something they are (a fingerprint). By requiring not only the Master Password, but also an additional login factor (like a one-time password, a fingerprint swipe, or a randomly-generated 6-digit code), a user adds another layer of protection against unauthorized access to their LastPass account.

If an attacker were to discover a user's Master Password, it's unlikely that they would also have access to a valid multifactor token, therefore minimizing the chance that they would be able to gain access to the user's LastPass account.

Admins can mandate multifactor authentication through policies in the admin dashboard, requiring use of any supported multifactor authentication option or requiring use of only specific, company-approved multifactor authentication options.

Shared Folders

Public Key Cryptography

LastPass uses RSA public key cryptography to allow users to share credentials with trusted parties synced through LastPass. Admins and users can create Shared Folders to give appropriate access to individuals or groups, without the need to expose the credentials themselves. And even though it is shared through LastPass, LastPass is unable to decrypt the data.

RSA uses asymmetric key algorithms, where the key used to encrypt a message is different from the key used to decrypt it. Each user has a pair of cryptographic keys, one public, one private. The public key can be shared with anyone and can be used to encrypt data, while the private key is available only to the user and can be used to decrypt data encrypted with their public key.

When a Shared Folder is created, a 256-bit encryption key is generated and used to encrypt the data stored in the Shared Folder. This encryption key is further encrypted with the public key of anyone invited to the Shared Folder and can be decrypted only with the invitee's corresponding private key.

All users who share folders generate a 2048-bit RSA key pair locally on their own device. The user's private key is encrypted with their vault encryption key using AES 256-bit encryption then sent to LastPass along with the user's public key. The encrypted private key is sent to LastPass so that it can be attained from other devices in the future. Public keys will be used by other users to encrypt data that can only be decrypted with the original private key.

Securing Shared Credentials

Because password sharing is inevitable in the workplace, ensuring the security of shared access to credentials is critical to maintaining compliance and safeguarding your company from threats. LastPass facilitates this in a way that provides convenient access for the employee, while maintaining accountability and tracking to meet your company's security requirements. Both private and work-related credentials can be shared through LastPass.

We strongly recommend using the password generator to create a unique, strong password for an account before sharing it.

When a user shares a credential, regardless of whether the password is "hidden" from the recipient in LastPass or made visible, the recipient can then launch the site that results in the autofill of the credentials. Once a shared credential is auto-filled on a website, it is outside of LastPass' control, and savvy end users may be able to obtain the password. For example, the recipient may have the capability to use the browser's developer tools to reveal the password.

Linked Accounts

LastPass users can link a personal account to their workplace account. By linking accounts, the user's personal vault is shared with their work account. This works in much the same way as Shared Folders. The difference is that this user has the Master Password for both vaults.

If a Super Admin activates the Super Admin Master Password Recovery feature, the linked account is automatically removed, ensuring admins do not have access to an employee's personal vault. The employee can re-link their personal vault when they've regained access to their work vault.

Account Recovery

Because LastPass does not store the Master Password, it does not offer the same password reset options users may be accustomed to on other web services. If a user forgets their Master Password, LastPass cannot look up the Master Password, reset the Master Password, or create a new Master Password for the user.

One Time Passwords (OTP) can be used for account recovery if a user's Master Password is lost. On desktop browsers, recovery OTPs are created automatically on a device when logging in. On mobile devices, users must opt-in to account recovery.

Login One Time Passwords

Users can generate and print Login OTPs on an ad hoc basis for use when logging in to LastPass on an untrusted device. A user can generate OTPs to log in on an untrusted computer in place of the Master Password, but they expire after first use.

Login OTPs work the same way as Recovery OTPs but, unlike a Recovery OTP, they remain enabled until used and are not stored on a local device. Using a Login OTP can safeguard against keylogging on public or untrusted computers. Learn more here (https://support.logmeininc.com/lastpass/help/use-temporary-one-time-passwords-lp030002).

Recovery

On desktop browsers, a random recovery key is generated on the user's device at login. This key is used to encrypt the vault encryption key (which is re-generated when logging in) using AES-128 in CBC mode. The encrypted vault key is then sent to LastPass servers while the recovery key is stored locally on the device.

On mobile devices, the recovery key is stored securely on the mobile device, behind biometrics. Account recovery is not available on mobile if the user does not have biometrics enabled on their device. If the user later disables biometrics, the recovery key is automatically deleted, and the recovery feature is disabled.

The encrypted vault key cannot be fetched from LastPass until account recovery is activated by the user. The encrypted vault key on LastPass' servers is secure because the recovery key is never shared with LastPass.

When Account Recovery is requested, a verification code is emailed or sent via a text message (in case of desktop) or sent via push notification (in the case of mobile). The user's identity is confirmed via access to the email account or phone number associated with the user's account, and biometrics in the case of mobile account recovery. After verification, the encrypted vault key is downloaded and decrypted locally on the user's computer using the recovery key. The user specifies a new Master Password, generates a new vault encryption key, a new login hash, and then encrypts their vault data with the new key. The old encrypted files are wiped from LastPass servers, thus invalidating the old keys.

If for any reason the OTPs are not available, whether because of a software or system upgrade, or because the user does not have access to a previously-used device, then the only recourse is to delete the account to start over. LastPass cannot do anything in this case to recover the encrypted data or reset the Master Password, because that recovery data is only available client-side rather than server-side. Again, we've designed LastPass this way as a protective measure to reduce the risk of someone maliciously obtaining a user's sensitive data.

Account recovery OTPs can be disabled by the user, or disabled organization-wide with a LastPass security policy. In addition, account recovery OTPs are immediately invalidated when a user changes their Master Password, and new OTPs will be generated when the user successfully logs in again on desktop (or through re-enabled account recovery on mobile).

A super admin security policy is available to admins, allowing designated admins to reset the Master Password of employee accounts.

Securing the Client

The LastPass client is typically run as a browser extension that is supported for all major browsers on Windows, Mac, and Linux. Native applications are also available for iOS, Android, Windows, and OSX.

Communication between the client and LastPass servers uses TLS connections.

Connections via browser extensions are further protected by browser security controls. HTTP Strict Transport Security (HSTS) forces all connections to TLS, thus mitigating the risks of downgrade attacks and misconfiguration. Content Security Policy headers provide further protection from injection attacks, such as cross-site scripting.

LastPass Federated Login Services

Introduction

The architectural design of LastPass Federated Login Services maintains the zero-knowledge model, even while the user authentication itself is performed by an identity provider (IdP) rather than directly by LastPass. LastPass Federated Login Services is designed to ensure that the user's IdP credentials are not exposed to LastPass and all data stored encrypted on LastPass' servers.

Federated Login with Cloud IdP (Microsoft Azure Active Directory)

Infrastructure Elements

LastPass cloud-based Federated Login Services uses the OpenID Connect and OAuth 2 implicit login flow. All sensitive information remains in the LastPass Clients and is never stored on a LastPass server unencrypted.

Service Provider (LastPass)

accounts.lastpass.com (ALP): A separate server where authentication data and half of the Hidden Master Password are stored.

Identity Provider (Microsoft Azure Active Directory)

Microsoft Azure Active Directory: Used as the directory service provider and identity provider. One half of the Hidden Master Password is stored as a user account attribute.

Creating the Hidden Master Password

By default, every LastPass user has a Master Password that is used to generate an encryption key for their LastPass vault, which is used to log into their LastPass account. In the case of federated login users, a **Hidden Master Password** is used to encrypt the vault data, and the Identity Provider is used to log into the LastPass account.

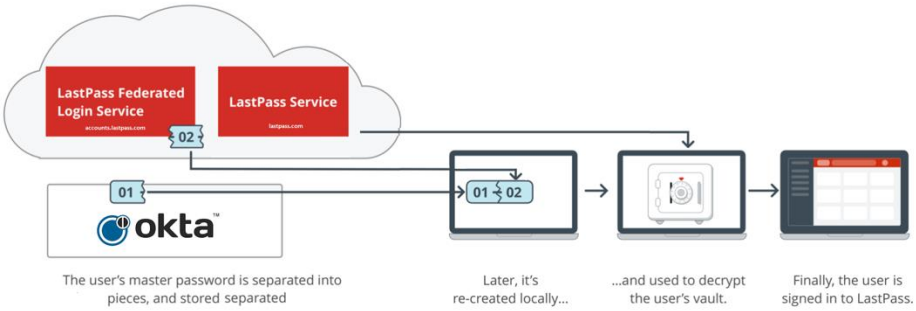
Provisioning is done through the LastPass Azure AD SCIM Integration and every new federated user will receive an invitation email with a temporary Master Password. This temporary Master Password is replaced during the first Azure AD login with randomly generated Hidden Master Password.

The Hidden Master Password is generated on the user's device from two crypto random generated strings (K1 and K2) and stored separately. One key is stored in the company's Azure AD and one is stored in accounts.lastpass.com. Together, the two keys (K1, K2) equal the user's Hidden Master Password, using the following algorithm:

Hidden Master Password = base64(SHA256(K1 XOR K2))

Federated login can be started in one of the LastPass Clients (e.g. LastPass Chrome Extension) by entering the email address. The user will then be redirected to Azure AD to authenticate. After a successful authentication, the two keys are returned to the LastPass Client and re-combined locally on the user's machine to form the "Hidden Master Password" which decrypts the LastPass vault.

Federated Login with Cloud IdP (Okta)



Infrastructure Elements

LastPass cloud-based federated login services uses the OpenID Connect and OAuth 2 implicit login flow. All sensitive information remains in the LastPass Clients and is never stored on a LastPass server unencrypted.

Service Provider (LastPass)

accounts.lastpass.com (ALP): A separate server where authentication data and half of the Hidden Master Password are stored.

Identity Provider (Okta)

Used the as the company's identity provider. One half of the Hidden Master Password is stored as a user account attribute.

Creating the Hidden Master Password

By default, every LastPass user has a Master Password that is used to generate an encryption key for their LastPass vault, which is used to log in to their LastPass account. In the case of federated login users, a **Hidden Master Password** is used to encrypt the vault data, and the Identity Provider is used to log in to the LastPass account.

Provisioning is done through the LastPass Okta SCIM Integration and every new federated user will receive an invitation email with a temporary Master Password. This temporary Master Password is replaced during the first Okta login with randomly generated Hidden Master Password.

The Hidden Master Password is generated from two crypto random generated strings (K1 and K2) and are stored separately. One part of the key is the company-wide secret key stored in Okta, and one generated on the user's device (K2) and stored in accounts.lastpass.com. Together, the two keys (K1, K2) equal the user's Hidden Master Password, using the following algorithm:

Hidden Master Password = base64(SHA256(K1 XOR K2))

Federated login can be started in one of the LastPass Clients (e.g. LastPass Chrome Extension) by entering the email address. The user will then be redirected to Okta to authenticate. After a successful authentication, the two keys are returned to the LastPass Client and re-combined locally on the user's machine to form the "Hidden Master Password" and decrypt the LastPass vault.

Federated Login with On-Premise IdP (Microsoft Active Directory Federation Services)

Infrastructure Elements

LastPass on-premise federated login services implements the Security Assertion Markup Language 2.0 (SAML 2.0) standard. As is typical with SAML, there are three main components that make up the integration, where LastPass is the Service Provider (SP), your organization acts as the Identity Provider (IdP), and the data is made available through a user agent (the browser or app).

For LastPass Federated Login Services, the primary infrastructure components are:

Service Provider (LastPass)
lastpass.com (LP): The web and database servers where the encrypted vault and one third of the Master Password are stored.

accounts.lastpass.com (ALP): A separate server where authentication data and one third of the Master Password are stored.

Identity Provider (Company's Active Directory Infrastructure)

Active Directory: Hosted by the company and serves as the directory service provider. One third of the Master Password is stored as a user account attribute.

Active Directory Federation Services: Hosted by the company and authenticates the user as they log in to LastPass.

ADFS Custom Attribute Store: Provides a secure way for adding the encrypted K1 to the SAML response.

LastPass Active Directory Connector: A LastPass client run by the company to sync user status from their Active Directory to LastPass.

User Agent (Browser/LastPass Extension)

Creating the Master Password

By default, every LastPass user has a Master Password that is used to generate an encryption key for their LastPass vault and is used to log in to their LastPass account.

With LastPass Federated Login Services, the Master Password is created behind-the-scenes by the LastPass AD Connector. As a user is provisioned, the LastPass AD Connector generates three 256-bit keys using a cryptographically secure, pseudo-random number generator.

Together, the three keys (K1, K2, and K3) equal the user's Master Password, using the following algorithm:
MasterPassword = base64(SHA256(K1 XOR K2 XOR K3))

With the Master Password, the LastPass AD Connector generates a vault encryption key that is used to encrypt the user's newly-created vault. The encrypted vault and the login hash are then sent to lastpass.com to complete the user provisioning step.

K1, K2, and K3 are then stored separately, one in the company's Active Directory, one in accounts.lastpass.com, and one in lastpass.com.

As the user logs in to LastPass, they are redirected to ADFS to authenticate. After successfully authenticating, the three keys are re-combined locally on the user's machine to form the "Master Password" and decrypt the LastPass vault.

The three keys – K1, K2, and K3 – are only recombined locally on the user's device to form the Master Password and decrypt the vault. If the company-wide key is selected, all users will share the same K1.

SOC 2 Attestation

LastPass has acquired the Service Organization Control 2 (SOC 2) Type II attestation report and a SOC 3 report from an industry-recognized auditing firm.

The SOC 2 report provides a rigorous audit, review, and validation of our practices as a company, and with respect to our services and systems, around data security, safeguarding of information, and service availability. This includes ensuring proper safeguarding of the data our systems process and the availability of those systems.

This includes control criteria related to:

- Security Monitoring
- Logical and Physical Access Controls
- System Operations

An annual review must be completed to maintain SOC 2 compliance. The SOC 2 Type II assessment also validated that LogMeIn meets the objectives of the Cloud Computing Compliance Control Catalogue (C5) requirements.

The SOC 2 Type II attestation confirms that LastPass security controls are well designed and operating effectively.

As the "gold standard" for software companies that is widely recognized nationwide across industries, completing and maintaining the SOC 2 attestation is just one more way we demonstrate our commitment to security and privacy.

As evidence of SOC 2 compliance, SOC 3 reports are publicly available (https://www.logmeininc.com/trust). An attestation of the SOC 2 plus German C5 Compliance report is also available upon request in both English and German. The German C5 Compliance Standard is promoted by the German Federal Office for Information Security which produces the criteria for the C5 based on the international ISAE 3000 standard.

LastPass Infrastructure

High-Availability Service

LastPass is built with fully redundant data centers, reducing the risk of downtime and single-point-of-failure.

Even if a user does not have internet access, they can still access their account via the LastPass browser extension or app on a device where they have previously logged in. A secure, local copy of a user's vault content is stored automatically when a user connects to LastPass, which is then available offline.

The status of the LastPass service is currently reported here (https://status.lastpass.com/).

Service Architecture

LastPass operates in two data centers in the United States and two in Europe. All data centers are in world-class hosting facilities that constantly monitor environmental conditions and provide 24-7 physical security. User vault data is backed up daily. By default, when a new LastPass account is created, vault data is stored in the United States, and Account Data is stored and replicated in our data centers in the United States and Europe. Business account holders can request that their vault data be stored locally in Europe, Australia, Singapore, or India instead of the United States.

- LastPass also leverages public cloud services for specific functionality.
- Attachments are stored in AWS S3.
- Additional Azure services are used by LastPass SSO and LastPass MFA services.

Automated nightly reviews are conducted to ensure the appropriate level of security.

Universal Availability

LastPass strives to offer users access to their data on as many platforms as possible and keeps pace with new technology so that users can always rely on LastPass to securely sync their data where they need it. Information on supported platforms, browsers, and mobile devices can be found here (https://support.logmeininc.com/lastpass/help/system-requirements-for-users-lp010008).

The LastPass web vault is also available at www.LastPass.com on all major browsers and platforms. Although downloading the extensions and apps are recommended for the best experience, the web vault ensures secure access on devices where the LastPass client is not installed.

Local and Cloud Storage

To ensure that customers have consistent access to their data, LastPass creates an encrypted copy of the vault both locally on a user's device and in the cloud on LastPass' servers.

Protecting Data at Rest

When using the LastPass browser extensions or the LastPass mobile apps, LastPass stores a locally-encrypted, cached copy of the vault on that device. If LastPass.com can't be reached because the user has no internet connection or in the unlikely event that LastPass.com is down, the user can log in via the browser extension or the mobile app to access the stored data.

The secure offline cache is only available if the user has successfully logged in to the extension or mobile app at least once before to sync with the LastPass servers.

On Windows devices, Windows Crypto APIs are used to add an extra layer of protection.

Note that offline access can be disabled in the LastPass extension preferences or disabled company-wide with a LastPass security policy.

User Data Storage

Sensitive vault data is encrypted client-side, then received and stored by LastPass as encrypted data. Other data, such as a phone number used for SMS account recovery, is encrypted server-side using a Hardware Security Module (HSM). The HSM is a separate device purposely built to securely store cryptographic keys.

Login Hash Storage

LastPass receives the login hash from the user (following the default 100,101 iterations on the user's Master Password using PBKDF2-SHA256), the login hash is additionally salted with a random 256-bit salt, and an additional 100,000 rounds of PBKDF2-SHA256 are performed. That output is then hashed using

scrypt to increase the memory requirements of brute-force attacks. The resulting hash stored by LastPass is the output of 200,101 rounds of SHA256 + scrypt.

LastPass System Data

EncFS is used to encrypt system data needed to run the LastPass service. EncFS is a Filesystem in Userspace (FUSE)-based encrypted filesystem that automatically encrypts all files added to the volume. A system administrator is required to manually enter the password to decrypt the filesystem.

Transport Layer Encryption

LastPass uses TLS exclusively for secure data transfer even though the vast majority of user data is already encrypted with AES 256. This protocol protects the data from any party listening in to the network traffic. TLS ensures that the user is connecting directly to LastPass to protect against man-in-the-middle attacks.

Protecting LastPass Network and Systems

LastPass protects infrastructure and customer data with best practices and regularly-upgraded systems.

Application Layer Firewalls and Filtering

LastPass utilizes a best in class application firewall and DDoS prevention service. Traffic to LastPass services is proxied through this service, which filters and blocks malicious traffic before it reaches LastPass servers.

LastPass runs a local application firewall on its web servers to provide an additional layer of protection against web application attacks. This also actively blocks malicious traffic, such as SQL injection and XSS (cross-site scripting) attacks.

Network Layer Firewalls and Filtering

All LastPass web servers are running host-based firewalls which filter inbound and outbound connections including internal connections between LastPass systems. Only ports 80 and 443 are open to the internet.

Vulnerability Testing

Vulnerability scans are run daily against LastPass servers. LastPass also uses automated tools to search for common mistakes that could result in vulnerabilities such as XSS or SQL Injection.

Third-Party Audits

We're committed to evaluating and improving LastPass through third-party audits and penetration tests, and LastPass infrastructure is tested by an industry-recognized third party on an annual basis.

Error Reporting

LastPass may also collect anonymized error reports and crash data from users to help us continually improve the service. Although users can opt-out of this when installing LastPass, no identifying information is used in these automated error reports, which are solely used by the LastPass team to improve performance and security.

Code Reviews

All changes to the code base are reviewed by the technical team for security, privacy, and compliance with company policies and procedures.

LastPass MFA Solution Security

Though the MFA solution shares many core components with the overall LastPass infrastructure, we want to specifically address how biometric data is used and stored and provide further information on service architecture and infrastructure that are unique to the MFA solution.

Components

The LastPass MFA solution includes the mobile apps for iOS and Android, Desktop apps for Windows and Mac OS, the Management Console, and the LastPass servers.

Application Security

LastPass MFA does not store users' credentials - such as user's biometric or passwords - on its authentication server or any centralized database. Instead it generates and encrypts a mathematical representative of biometrics through a one-way irreversible process. Then it encrypts and locally stores a mathematical representative of the biometric data locally on the user's device, and discards raw biometric data. The solution only leverages mathematical representations of biometrics on the user's device and doesn't transfer any data to the server. Handling sensitive data in this way diminishes the feasibility of many scalable attacks that could compromise user information.

LastPass uses techniques to detect if the application has been modified, reacts to the attack by deleting its sensitive information, and alerts the LastPass servers. This will be the best course of action to prevent hacking attempts that try to circumvent code protection or modify the code in run time or at rest.

FIDO Compliance

LastPass is a FIDO UAF certified solution that goes beyond FIDO requirements and even PKI to make sure the authentication is processed securely and privately. As a FIDO member, we believe FIDO is a good start, but it does not enforce adequate security requirements needed to address today's enterprise cyber-security challenges.

Biometric Data Usage

LastPass MFA offers Human and Hidden Multifactor Authentication (H2MFA) to more securely identify and authenticate users. When a user authenticates, they supply biometrics (such as fingerprint). Then, LastPass extracts hidden information from the user's phone and combines it with the Public Key Infrastructure (PKI) to verify and grant access to the user.

"Human factors" refer to biometric data that is collected by the user via their smartphone. The following biometrics can be utilized by LastPass MFA:

- Fingerprint
- Face
- Pattern

"Hidden factors" refer to data that is generated by and unique to a user's phone, and includes:

Signed Challenge (Public Key Infrastructure - PKI)
Used for encryption and for signing an authentication challenge that a device receives. Every user has a private key generated during the registration process. The app signs the challenge received from the server with the user's private key and sends that to the server. When the user needs to be authenticated, authentication servers send an authentication request to the user's phone. This authentication request includes an RSA cryptography challenge. If the user accepts the request, and is successfully authenticated based on their biometrics, then the app signs the challenge with the user private key. The server also has a public key version of the user's private key. Once the server receives the authentication response from the app, after decrypting all the hidden factors, it also checks the signed challenge received from the app to verify the authenticity of the signed challenge.

LTOTP (Location and time-based one-time passcode)
A long randomly generated passcode that changes based on time and location. Must have algorithm and token (1 copy of token is on the server, one is on the phone) and a new token based on time and location of the user is generated every time a user tries to reauthenticate.

Location
Pulls in GPS and WIFI information based on the user's location.

Hardware signature
A device-specific ID that identifies the unique phone.

Adaptive and Contextual Authentication
The items that are shown to the user are helping LastPass MFA do adaptive authentication. LastPass MFA learns about the user's device, time, and location, then uses those factors to determine the level of risk. The more LastPass MFA is used, the smarter the requests will get.

Biometric Data Storage

When it comes to iris and fingerprint information, mobile phone manufacturers manage the information and is not accessible by LastPass or any other third-party applications. In this case, users' biometrics are managed by mobile device manufacturers and are securely stored on the hardware element of the phone, designed for storing sensitive information such as fingerprint.

When it comes to native biometrics on mobile such as fingerprint, iris, or native face authentication such as FaceID, LastPass converts the biometric raw data to mathematical models, and then discards the raw data after the enrollment process. Converting biometric raw data to mathematical models is a one-way process that cannot be reversed. Therefore, it is not possible to extract biometric raw data from the mathematical model of biometrics. The mathematical model of biometrics is then encrypted using the AES 256 cryptography method. The encrypted version of the mathematical model is then stored in the app sandbox and the encryption key along with communication keys are stored in secure enclave in iOS devices or Trusted Environment of TEE in Android devices.

LastPass mobile apps and SDKs take advantage of white-box cryptographic technique which enables LastPass to protect sensitive information, the mathematical models of biometrics, and the cryptographic keys at rest and during runtime, even if the mobile device is encrypted by a hardware module for storing sensitive information. For Android, LastPass code protection goes beyond the typical bytecode obfuscators converting Java bytecode to obfuscate native code which make decompiling bytecode impossible.

It is important to emphasize that throughout the LastPass solution, no information is stored unencrypted. All of the user's sensitive information is encrypted and secured on the user's phone. At no point does the biometric information of the user or the mathematical model of the biometrics leave the user's phone. The user's sensitive information is always under their control and at any point they can delete this information from their phone.

Also, LastPass MFA code protection techniques obfuscate the code base, inject hundreds of overlapping integrity checkers, and embed platform specific anti-debug, anti-piracy and anti-malware code. White-box cryptography techniques are used to protect LastPass mobile applications and SDKs against hackers. Sensitive parts of LastPass applications are run through white-box cryptography environment to make sure all sensitive information is secured in run time and at rest. All cryptographic keys are secured in a secure key box container that allows the algorithm to operate directly on encrypted keys. This means keys are not revealed by debugging and reverse engineering.

Measuring Accuracy

All biometric authentication methods supported by LastPass have a maximum false acceptance rate of 0.002%. This means to get one false match, an imposer would on average need to provide 50,000 samples to the LastPass app. Since comparison of biometric authentication is done locally on user's device, it's impractical to attempt thousands (let alone tens or hundreds of thousands) of break-ins, as an intruder will need direct access to the user's phone. In addition, LastPass can limit the number of tries to 3, 5, or 9 (depending on the settings enabled). Built-in quality analysis rejects low-quality images and recordings, further discouraging spoofing.

LastPass includes anti-spoofing features, liveliness and behavioral tests, and additional authentication to ensure the biometric information received is from a real live user with their unique behavioral and environmental factors.

Lost or Stolen Phone

Possession of a user's phone does not simply enable an intruder to access the user's account. The intruder would need to also spoof the user's biometrics and behaviors, which are very costly and time-consuming processes. To successfully access a user's account, the intruder needs to first steal the user's phone, break the phone lock if active, then perform a spoofing attack.

Once a user realizes that they have lost their phone, they can report it to their IT team or to LastPass. As soon as it is reported, all access to LastPass MFA on the lost or stolen phone is blocked and all the user's data contained therein is deleted remotely from the phone. LastPass also provides self-service solutions through which a user can go online and report their phone as lost, subsequently, blocking access from that phone and deleting their information.

Security Incident Reporting

Security is our highest priority at LastPass, including quickly responding to and fixing reports of bugs or vulnerabilities. LastPass is in part able to achieve the highest level of security for our users by looking to our community to challenge our technology.

LastPass classifies security reports into two categories: a concern reported by a security researcher and a concern reported by a LastPass customer.

Bug Bounty Program

LastPass participates in a bug bounty program (https://bugcrowd.com/lastpass) hosted at BugCrowd to facilitate the important work that security researchers do to find and responsibly disclose qualifying security bugs. We appreciate the important work that the security research community provides and appreciate responsible disclosure of issues; we believe that when the security process works as designed, we all benefit.

Please note, we only accept reports through Bug Bounty for our Password Manager currently. We will soon accept reports for our SSO and MFA solutions as well.

A report by a security researcher should be reported through our bug bounty program at https://bugcrowd.com/lastpass.

Security Concerns Reported by Customers

Customers with a security concern should report it in a support ticket (https://support.logmeininc.com/lastpass/help/how-do-i-contact-customer-support-for-lastpass-lp010121) where it will be escalated appropriately.

Reporting Security Issues

When reporting potential issues, we ask that users please try to be as thorough as possible in providing us enough information so that we can appropriately recreate their findings.

This may include exact steps to reproduce the bug, any links that were clicked on, pages that were visited, URLs, and any affected account email addresses. Please include a code sample and either images or a video recording that clearly demonstrates the exploit.

If using automated tools to find vulnerabilities, please be aware that these tools frequently report false positives.

Responding to Security Concerns

Once a security concern has been submitted and received, our team will:

1. Promptly take steps to investigate the report and determine its severity.
2. Contact the reporter directly if more information is needed.
3. Try to fix the issue or perform a best effort at mitigation. While issues are usually fixed quickly, deploying the fix to affected customers will be done based on the complexity and severity of the issue.
4. Once the issue is fully resolved to both the reporter’s and our satisfaction, we'll close the report.